

Universidad de La Salle

Facultad de Ingeniería

Programa de Ingeniería de Software

2025-2_LÓGICA DE PROGRAMACIÓN II G_01

ACTIVIDAD 2: Pongamos a prueba nuestros conocimientos sobre el lenguaje Java

Presentado por:

Andriely Alejandro Casallas Calderon

Juan Felipe Rodríguez Castellanos

Presentado a:

Jhon Alexis Méndez Lara

Docente virtual

Bogotá D.C., Colombia

4 de agosto de 2025

Introducción

En este documento se van a presentar las soluciones que planteamos para la **Actividad 2** del curso **2025-2_Lógica de Programación II G_01**, si desea descargar el código fuente puede verlo accediendo al repositorio que se encuentra en el siguiente enlace de GitHub:

[Repositorio Online](#)

EJERCICIO 1:

Este código lee una nota y muestra el mensaje correspondiente según el rango de calificación.

```
/* Código realizado por Juan Castellanos y Andriely Casallas 2025
 * Programa que simula un concurso de baile donde cuatro jueces califican a
 * Los participantes.
 * El programa calcula el puntaje final de cada participante excluyendo la
 * calificación más baja.
 *
 * Realizado con fines educativos para La universidad La Salle.
 */

import java.util.Scanner;

// Definición de la clase Estudiante
public class Estudiante {
    // Scanner para entrada de datos por consola
    Scanner input = new Scanner(System.in);
    // Nombre del estudiante
    String nombre;
    // Nota del estudiante
    double nota;

    // Constructor de la clase Estudiante
    public Estudiante(String nombre, double nota) {
        this.nombre = nombre; // Asigna el nombre recibido
        this.nota = nota;     // Asigna la nota recibida
    }
}
```

```

/* Código realizado por Juan Castellanos y Andriely Casallas 2025
 * Programa que simula un concurso de baile donde cuatro jueces califican a
Los participantes.
 * El programa calcula el puntaje final de cada participante excluyendo la
calificación más baja.
*
* Realizado con fines educativos para la universidad La Salle.
*/
import java.util.Scanner;

public class Main {
    public static void main(String[] args) throws Exception {
        Scanner input = new Scanner(System.in); // Scanner para leer datos
del usuario
        boolean condition = true; // Controla el ciclo principal del
programa
        do {
            String nombre = "";
            double nota = 0.0;
            boolean banderaNombre = true; // Controla la validación del
nombre
            boolean banderaNota = true; // No se usa en el código actual
            Estudiante estudiante = new Estudiante(nombre, nota); // Crea un
objeto Estudiante
            do {
                // Solicita el nombre del estudiante
                System.out.println("Hola, bienvenid@ al sistema de notas.");
                System.out.println("Por favor, ingrese el nombre del
estudiante:");
                nombre = input.nextLine();
                estudiante.nombre = nombre;
                // Valida que el nombre no esté vacío, tenga solo letras y
sea mayor a 7 caracteres
                if (!nombre.isEmpty() && nombre.matches("[a-zA-
ZáéíóúÁÉÍÓÚñüÜ\\s]+") && nombre.length() > 7) {
                    banderaNombre = false;
                }
                else{
                    // Mensaje de error si el nombre no es válido
                    System.out.println("El nombre que ingresó no es válido.
Por favor, ingrese un nombre correcto.");
                }
            } while (!banderaNombre);
            nota = input.nextDouble();
            estudiante.nota = nota;
            System.out.println("Nota ingresada: " + nota);
            System.out.println("Nota almacenada: " + estudiante.nota);
            System.out.println("Nombre almacenado: " + estudiante.nombre);
            System.out.println("Nombre ingresado: " + nombre);
            System.out.println("Bandera Nombre: " + banderaNombre);
            System.out.println("Bandera Nota: " + banderaNota);
            System.out.println("Condición: " + condition);
            System.out.println("Input: " + input);
            System.out.println("Scanner: " + scanner);
        } while (condition);
    }
}

```

```

        System.out.println("=====");
        =====");
        continue;
    }
} while (banderaNombre == true);

// Sigue la nota del estudiante
System.out.println("Por favor, ingrese la nota del estudiante
(0.0 a 10.0): ");
nota = input.nextDouble();
input.nextLine(); // Limpia el buffer
estudiante.nota = nota;
// Prepara la información para mostrar
String InformacionCompleta = "Información registrada: \n" +El
estudiante "+ estudiante.nombre + " con nota: " + estudiante.nota;
String otroEstudiante = "¿Desea ingresar otro estudiante? (s/n): ";
;

// Clasifica la nota y muestra el resultado
if (nota >= 0 && nota <= 5) {
    System.out.println("Suspensos");
} else if (nota > 5 && nota <= 6.5) {
    System.out.println("Aprobado");
} else if (nota > 6.5 && nota <= 8.5) {
    System.out.println("Notable");
} else if (nota > 8.5 && nota < 10) {
    System.out.println("Sobresaliente");
} else if (nota == 10) {
    System.out.println("Matrícula de honor");
} else {
    // Mensaje de error si la nota no es válida
    System.out.println("Error en la calificación.");
    InformacionCompleta = "La nota ingresada no es válida. Por
favor, intente de nuevo con otro estudiante.";
    otroEstudiante = "¿Desea intentar de nuevo con otro
estudiante? (s/n): ";
}
// Muestra la información registrada
System.out.println(InformacionCompleta);
System.out.println("=====");
=====");
System.out.println("=====");
=====");
System.out.println(otroEstudiante);
// Pregunta si desea ingresar otro estudiante

```

```

        String bandera = input.nextLine();
        if (bandera.equals("s")) {
            condition = true;
        } else if (bandera.equals("n")) {
            System.out.println("Gracias por usar el sistema de notas,
                ¡¡Hasta luego colega!!.");
            condition = false;
        }
    else{
        // Mensaje si la opción no es válida
        System.out.println("Por favor, ingrese una opción válida
(s/n).");
    }
} while (condition); // Repite mientras el usuario lo desee
}
}

```

EJERCICIO 2:

Este código calcula la nota mínima necesaria en el tercer corte para aprobar y valida si ya aprobó o no puede aprobar.

```

// Clase que contiene la lógica del cálculo de notas
public class CalculoNota {

    // ===== Atributos =====
    double porcentajeCorte1;    // Porcentaje del primer corte
    double porcentajeCorte2;    // Porcentaje del segundo corte
    double porcentajeCorte3;    // Porcentaje del tercer corte
    double notaMinima;          // Nota mínima necesaria para aprobar

    // ===== Constructor =====
    public CalculoNota() {
        // Inicializamos los atributos con los valores del enunciado
        this.porcentajeCorte1 = 0.35;
        this.porcentajeCorte2 = 0.35;
        this.porcentajeCorte3 = 0.30;
        this.notaMinima = 3.0; // Generalmente, en universidades, se aprueba
        con 3.0
    }

    // ===== Métodos =====
}

```

```
// Método que calcula la nota mínima que el estudiante debe sacar en el
tercer corte
public double calcularNotaNecesaria(double nota1, double nota2) {
    // Se calcula la nota parcial con los porcentajes de los dos
    // primeros cortes
    double acumulado = (nota1 * porcentajeCorte1) + (nota2 * 
porcentajeCorte2);

    // Se calcula cuánto necesita en el tercer corte para llegar a la
    nota mínima
    double notaNecesaria = (notaMinima - acumulado) / porcentajeCorte3;

    return notaNecesaria;
}

// Método que evalúa la situación del estudiante y devuelve un mensaje
public String evaluarSituacion(double nota1, double nota2) {
    // Calculamos la nota necesaria en el tercer corte
    double notaNecesaria = calcularNotaNecesaria(nota1, nota2);

    // Si ya aprobó con los dos primeros cortes
    if (notaNecesaria <= 0) {
        return "Felicitaciones! Aprobaste el curso con los dos primeros
cortes";
    }

    // Si es imposible aprobar
    if (notaNecesaria > 5.0) {
        return "Ya no hay posibilidad que apruebes el curso. Debes
estudiar mejor para el próximo semestre";
    }

    // Si la nota necesaria es mayor a 3.0 (caso desafiante)
    if (notaNecesaria > 3.0) {
        return "Necesitas sacar al menos " + String.format("%.2f",
notaNecesaria) + " en el tercer corte para aprobar.";
    }

    // Si necesita 3.0 o menos, se considera que ya está prácticamente
    // aprobado
    return "Felicitaciones! Aprobaste el curso con los dos primeros
cortes";
}
```

```
import java.util.Scanner;
import java.util.Locale;

public class Main {
    public static void main(String[] args) {
        // Configurar Scanner para que use coma como separador decimal (ej: 4,3)
        Scanner sc = new Scanner(System.in);
        sc.useLocale(Locale.forLanguageTag("es-ES"));

        // Crear objeto de la clase CalculoNota
        CalculoNota calculo = new CalculoNota();

        // ====== Solicitar nombre completo del estudiante ======
        System.out.print("Ingrese su nombre completo: ");
        String nombre = sc.nextLine();

        double nota1, nota2;

        // ===== Validación de la nota 1 =====
        System.out.print("Ingrese la nota del primer corte (0,0 - 5,0): ");
        nota1 = sc.nextDouble();
        while (nota1 < 0.0 || nota1 > 5.0) {
            System.out.println("⚠ Error: La nota debe estar entre 0,0 y 5,0.");
            System.out.print("Ingrese nuevamente la nota del primer corte: ");
            nota1 = sc.nextDouble();
        }

        // ===== Validación de la nota 2 =====
        System.out.print("Ingrese la nota del segundo corte (0,0 - 5,0): ");
        nota2 = sc.nextDouble();
        while (nota2 < 0.0 || nota2 > 5.0) {
            System.out.println("⚠ Error: La nota debe estar entre 0,0 y 5,0.");
            System.out.print("Ingrese nuevamente la nota del segundo corte: ");
            nota2 = sc.nextDouble();
        }

        // Evaluar situación con el método de la clase CalculoNota
```

```

        String resultado = calculo.evaluarSituacion(notas1, notas2);

        // Imprimir el mensaje con el nombre del estudiante
        System.out.println("\n==== RESULTADO ====");
        System.out.println("Estudiante: " + nombre);
        System.out.println(resultado);

        // Cerrar Scanner
        sc.close();
    }
}

```

EJERCICIO 3:

Este código calcula el promedio de las tres mejores calificaciones de cuatro jueces, validando que todas sean correctas.

```

/* Código realizado por Andriely Casallas y Juan Castellanos 2025
 * Programa que simula un concurso de baile donde cuatro jueces califican a
 * los participantes.
 * El programa calcula el puntaje final de cada participante excluyendo la
 * calificación más baja.
 *
 * Realizado con fines educativos para la universidad La Salle.
 */

public class Concurso {
    // Arreglo para almacenar las calificaciones de los jueces
    private double[] numeros;

    // Constructor: inicializa el arreglo de calificaciones con tamaño 4
    public Concurso() {
        numeros = new double[4];
    }

    // Método para establecer una calificación en una posición específica
    public void establecerNota(int indice, double valor) {
        if (indice >= 0 && indice < numeros.length) {
            numeros[indice] = valor;
        } else {
            System.out.println("Índice fuera de rango");
        }
    }
}

```

```

}

// Método para sumar todas las calificaciones
public double sumarNumeros() {
    double suma = 0;
    for (double num : numeros) {
        suma += num;
    }
    return suma;
}

// Método para obtener la calificación más baja
public double obtenerMenor() {
    double menor = numeros[0];
    for (double num : numeros) {
        if (num < menor) {
            menor = num;
        }
    }
    return menor;
}

// Método para calcular el promedio excluyendo la calificación más baja
public double sumarExcluyendoMenor() {
    double suma = sumarNumeros(); // Suma total de las calificaciones
    double menor = obtenerMenor(); // Calificación más baja
    double sumaComp = suma -= menor; // Suma excluyendo la más baja
    double promedio = sumaComp / 3; // Promedio de las tres
    calificaciones restantes
    return promedio;
}
}

```

```

/* Código realizado por Juan Castellanos y Andriely Casallas 2025
* Programa que simula un concurso de baile donde cuatro jueces califican a
los participantes.
* El programa calcula el puntaje final de cada participante excluyendo la
calificación más baja.
*
* Realizado con fines educativos para la universidad La Salle.
*/

import java.util.Scanner;

```

```
public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in); // Se crea el objeto Scanner
        para Leer la entrada del usuario
        Concurso concurso = new Concurso(); // Se crea una instancia de la
        clase Concurso
        boolean bandera = true; // Variable para controlar el ciclo
        principal

        do {
            // Mensaje de bienvenida y explicación
            System.out.println("===== (°|_|°) =====");
            System.out.println("Bienvenido al concurso de baile. Ingrese las
            puntuaciones de los jueces (entre 0 y 10):");

            // Ciclo para ingresar Las puntuaciones de Los 4 jueces
            for (int i = 0; i < 4; i++) {
                System.out.print("Ingrese el número " + (i + 1) + ": ");
                int valor = sc.nextInt(); // Se Lee La puntuación

                // Validación de La puntuación
                if (valor < 0 || valor > 10) {
                    System.out.println("Por favor, ingrese un número entre 0
y 10.");
                    i--; // Si La puntuación no es válida, se repite La
                    iteración
                    continue;
                } else {
                    concurso.estabNumero(i, valor); // Se almacena La
                    puntuación en el objeto Concurso
                }
            }

            // Se muestran Los resultados: menor puntuación y promedio
            // excluyendo el menor
            System.out.println("El número menor es:\n" +
            concurso.obtenerMenor() +
            "\nEl promedio de los números excluyendo el menor es: \n" +
            + concurso.sumarExcluyendoMenor());

            // Se pregunta al usuario si desea calcular el puntaje de otro
            participante
```

```

        System.out.print("¿Desea calcular el puntaje de otro
participante? (s/n): ");
        String respuesta = sc.next();
        if (respuesta.equalsIgnoreCase("n")) {
            bandera = false; // Finaliza el ciclo
            System.out.println("Gracias por usar el programa. ¡Hasta
luego!");
        } else if (respuesta.equalsIgnoreCase("s")) {
            bandera = true; // Continúa el ciclo
        } else {
            System.out.println("Respuesta no válida. Saliendo del
programa.");
            bandera = false; // Finaliza el ciclo por respuesta inválida
        }

    } while (bandera == true); // El ciclo se repite mientras bandera
sea true
}
}

```

EJERCICIO 4:

Este código cuenta cuántos números son pares, cuántos son impares y calcula el promedio de todos.

```

import java.util.ArrayList;
import java.util.List;

// Esta clase se encarga de hacer todo el trabajo "Lógico":
// guardar los números, contar cuántos pares/impares hay
// y calcular el promedio.
public class AnalizadorNumeros {

    // ===== Atributos =====
    // Usamos una lista para guardar todos los números que el usuario
ingrese.
    // La ventaja de una lista es que podemos guardar tantos números como
necesitemos
    // sin preocuparnos de cuántos son.
    List<Integer> numeros;
}

```

```
// ===== Constructor =====
public AnalizadorNumeros() {
    // Aquí inicializamos la lista vacía. Cada vez que el usuario
    // ingrese un número, lo vamos a ir metiendo en esta lista.
    this.numeros = new ArrayList<>();
}

// ===== Métodos =====

// Este método recibe un número y lo guarda en la lista.
// Básicamente es como ir "apilando" todos los números que da el
usuario.

public void agregarNumero(int numero) {
    numeros.add(numero);
}

// Este método cuenta cuántos de los números guardados son pares.
// Recorremos la lista uno por uno, y si el número es divisible entre 2
// (o sea, residuo cero), lo contamos como par.

public int contarPares() {
    int conteo = 0;
    for (int num : numeros) {
        if (num % 2 == 0) {
            conteo++;
        }
    }
    return conteo;
}

// Igual que el anterior, pero ahora verificamos cuáles son impares.
// Si al dividir entre 2 sobra algo (residuo distinto de cero), entonces
es impar.

public int contarImpares() {
    int conteo = 0;
    for (int num : numeros) {
        if (num % 2 != 0) {
            conteo++;
        }
    }
    return conteo;
}

// Calculamos el promedio de todos los números.
// Para eso sumamos todo y luego dividimos entre la cantidad total de
números.
```

```

public double calcularPromedio() {
    if (numeros.isEmpty()) {
        // En caso de que no haya números, devolvemos 0 para evitar
        error.
        return 0;
    }

    int suma = 0;
    for (int num : numeros) {
        suma += num; // Vamos sumando cada número.
    }

    // Convertimos a double para que el resultado sea decimal, no
    entero.
    return (double) suma / numeros.size();
}
}

```

```

import java.util.Scanner;

// Esta es la clase principal, donde corre el programa.
// Aquí le pedimos los números al usuario y mostramos los resultados
finales.
public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        // Creamos un objeto de nuestra clase AnalizadorNumeros.
        // Este objeto se encargará de guardar y analizar los números.
        AnalizadorNumeros analizador = new AnalizadorNumeros();

        // Primero preguntamos cuántos números quiere ingresar el usuario.
        // Con esto sabremos cuántas veces pedir un número.
        System.out.print("Ingrese la cantidad de números enteros a leer: ");
        int cantidad = sc.nextInt();

        // Ahora pedimos cada número, uno por uno, según la cantidad que nos
        dijeron.
        // Usamos un ciclo for porque sabemos exactamente cuántas veces
        vamos a repetir.
        for (int i = 1; i <= cantidad; i++) {
            System.out.print("Ingrese el número " + i + ": ");
        }
    }
}

```

```

        int num = sc.nextInt();
        // Cada número que el usuario da lo guardamos en nuestro
        analizador.
        analizador.agregarNumero(num);
    }

    // Ya que tenemos todos los números, ahora mostramos los resultados.
    // Para eso simplemente llamamos a los métodos que creamos en la
    otra clase.
    System.out.println("\n==== RESULTADOS ====");
    System.out.println("Total de números pares: " +
analizador.contarPares());
    System.out.println("Total de números impares: " +
analizador.contarImpares());
    System.out.printf("Promedio de todos los números: %.2f%n",
analizador.calcularPromedio());

    // Cerramos el Scanner para liberar memoria (buena práctica).
    sc.close();
}
}

```

EJERCICIO 5:

Este código valida la contraseña en máximo tres intentos y muestra un mensaje de acceso o bloqueo.

```

/* Código realizado por Andriely Casallas y Juan Castellanos 2025
 * Programa que simula un concurso de baile donde cuatro jueces califican a
los participantes.
 * El programa calcula el puntaje final de cada participante excluyendo la
calificación más baja.
*
* Realizado con fines educativos para la universidad La Salle.
*/
import java.util.Scanner;

public class Main {
    public static void main(String[] args) throws Exception {
        // Crear un objeto Scanner para leer la entrada del usuario

```

```

Scanner input = new Scanner(System.in);
// Crear un objeto Usuario para verificar la contraseña
Usuario usuario = new Usuario();
// Permitir hasta 3 intentos para ingresar la contraseña
for (int i = 0; i < 3; i++) {
    System.out.print("Ingrese la contraseña: ");
    // Leer la contraseña ingresada por el usuario
    String contraseniaIngresada = input.nextLine();
    // Verificar si la contraseña es correcta
    if (usuario.verificarContrasenia(contraseniaIngresada)) {
        System.out.println("¡Felicitaciones! Lograste ingresar.");
        break; // Salir del ciclo si la contraseña es correcta
    } else {
        // Mostrar mensaje de error y cantidad de intentos restantes
        System.out.println("Contraseña incorrecta. Intente de nuevo.
le quedan " + (2 - i) + " intentos.");
    }
}
// Mensaje mostrado cuando se exceden los intentos permitidos
System.out.println("=====");
System.out.println("Usted ha excedido el número máximo de intentos.
¡Por favor, intente más tarde!");
}
}

```

```

/* Código realizado por Andriely Casallas y Juan Castellanos 2025
 * Programa que simula un concurso de baile donde cuatro jueces califican a
los participantes.
 * El programa calcula el puntaje final de cada participante excluyendo la
calificación más baja.
*
* Realizado con fines educativos para la universidad La Salle.
*/
public class Usuario {

    // Almacena La contraseña válida para el usuario
    private String contrasenia = "Logica2021";

    // Verifica si la contraseña ingresada coincide con La contraseña
almacenada.
}

```

```

public boolean verificarContrasenia(String contraseniaIngresada) {
    // Compara la contraseña ingresada con la almacenada
    if (contraseniaIngresada.equals(contrasenia)) {
        return true;
    } else {
        return false;
    }
}

```

EJERCICIO 6:

Este código clasifica 10 personas en bebés, niños, adolescentes o adultos, validando que no haya edades negativas.

```

import java.util.ArrayList;
import java.util.List;

// Esta clase se encarga de manejar las edades que ingresa el usuario
// y clasificarlas en las categorías: bebés, niños, adolescentes y adultos.
public class ClasificadorEdades {

    // ===== Atributos =====
    List<Integer> edades; // Aquí vamos a guardar todas las edades

    // ===== Constructor =====
    public ClasificadorEdades() {
        this.edades = new ArrayList<>(); // Inicializamos la lista vacía
    }

    // ===== Métodos =====

    // Guardar una edad en la lista
    public void agregarEdad(int edad) {
        edades.add(edad);
    }

    // Contar cuántos son bebés (0 a 3 años)
    public int contarBebes() {
        int conteo = 0;
        for (int edad : edades) {
            if (edad >= 0 && edad <= 3) {
                conteo++;
            }
        }
    }
}

```

```

        }
    }
    return conteo;
}

// Contar cuántos son niños (4 a 12 años)
public int contarNinos() {
    int conteo = 0;
    for (int edad : edades) {
        if (edad >= 4 && edad <= 12) {
            conteo++;
        }
    }
    return conteo;
}

// Contar cuántos son adolescentes (13 a 17 años)
public int contarAdolescentes() {
    int conteo = 0;
    for (int edad : edades) {
        if (edad >= 13 && edad <= 17) {
            conteo++;
        }
    }
    return conteo;
}

// Contar cuántos son adultos (18 años o más)
public int contarAdultos() {
    int conteo = 0;
    for (int edad : edades) {
        if (edad >= 18) {
            conteo++;
        }
    }
    return conteo;
}
}

```

```

import java.util.Scanner;

// Clase principal: pide las edades al usuario y muestra el resultado

```

```
public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        // Creamos un clasificador, que hará el trabajo de contar cada grupo
        // de edad
        ClasificadorEdades clasificador = new ClasificadorEdades();

        System.out.println("Ingrese la edad de 10 personas:");

        // Vamos a pedir exactamente 10 edades
        for (int i = 1; i <= 10; i++) {
            System.out.print("Edad de la persona " + i + ": ");
            int edad = sc.nextInt();

            // Guardamos la edad en nuestro clasificador
            clasificador.agregarEdad(edad);
        }

        // Mostramos los resultados
        System.out.println("\n==== RESULTADOS ====");
        System.out.println("Bebés (0 a 3 años): " +
        clasificador.contarBebes());
        System.out.println("Niños (4 a 12 años): " +
        clasificador.contarNinos());
        System.out.println("Adolescentes (13 a 17 años): " +
        clasificador.contarAdolescentes());
        System.out.println("Adultos (18 años o más): " +
        clasificador.contarAdultos());

        sc.close(); // Cerramos el scanner
    }
}
```

EJERCICIO 7:

Este código calcula el promedio de lluvias por trimestre y el promedio total en un período de años usando ciclos anidados.

```
/* Código realizado por Juan Castellanos y Andriely Casallas 2025
 * Programa que simula un concurso de baile donde cuatro jueces califican a
Los participantes.
 * El programa calcula el puntaje final de cada participante excluyendo la
calificación más baja.
*
* Realizado con fines educativos para la universidad La Salle.
*/
```

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) throws Exception {

        boolean bandera = true; // Variable para controlar el ciclo
principal del programa
        do {
            // Mensaje de bienvenida
            System.out.println("Buenas tardes, te damos la bienvenida al
sistema Rainfall");
            Scanner input = new Scanner(System.in); // Crea un objeto
Scanner para Leer la entrada del usuario

            // Solicita al usuario la cantidad de años a registrar
            System.out.println("Por favor ingrese la cantidad de años que
quiere registrar: ");
            int anios = input.nextInt(); // Lee el número de años

            // Crea arreglos para almacenar la lluvia de cada trimestre por
año
            double [] lluvia1T = new double[anios];
            double [] lluvia2T = new double[anios];
            double [] lluvia3T = new double[anios];
            double [] lluvia4T = new double[anios];

            // Ciclo para cada año
            for (int i = 1; i <= anios; i++) {
                double lluviaTotalAnio = 0;
                // Ciclo para cada trimestre
                for (int j = 1; j <= 4; j++) {
```

```

        System.out.println("-----");
        System.out.println("A continuación ingrese por favor la
lluvia para el año: " + i + " y para el trimestre: " + j);
        double lluvia = input.nextDouble(); // Lee la cantidad
de lluvia para el trimestre

        // Validación: La lluvia no puede ser negativa
        if (lluvia < 0) {
            System.out.println("Error, la lluvia no puede ser
negativa, por favor ingrese un valor válido");
            j--; // Repite el trimestre si el valor es inválido
            continue;
        } else {
            // Guarda la lluvia en el arreglo correspondiente al
            trimestre
            switch (j) {
                case 1:
                    lluvia1T[i-1] = lluvia;
                    break;
                case 2:
                    lluvia2T[i-1] = lluvia;
                    break;
                case 3:
                    lluvia3T[i-1] = lluvia;
                    break;
                case 4:
                    lluvia4T[i-1] = lluvia;
                    break;
                default:
                    break;
            }
        }
    }

    // Calcula el promedio de lluvia para cada trimestre en todos
    Los años
    double promedio1T = calcularPromedio(lluvia1T);
    double promedio2T = calcularPromedio(lluvia2T);
    double promedio3T = calcularPromedio(lluvia3T);
    double promedio4T = calcularPromedio(lluvia4T);

    // Calcula la suma total de lluvia en todos los trimestres y
    años

```

```

        double totalSuma = suamTotal(lluvia1T) + suamTotal(lluvia2T) +
suamTotal(lluvia3T) + suamTotal(lluvia4T);
        double promedio = totalSuma / anios; // Promedio anual de Lluvia

        System.out.println("-----");
// Muestra el promedio de lluvia por año y Los valores de cada trimestre
        for (int i = 1; i <= anios; i++) {
            System.out.println("El promedio de lluvia para el año: "+ i +
" es de: "+ ((lluvia1T[i-1] + lluvia2T[i-1] + lluvia3T[i-1] + lluvia4T[i-
1])/4) +" pulgadas" );
            System.out.println("Cada trimestre del año: "+ i +" tuvo las siguientes lluvias: ");
            System.out.println("Trimestre 1: "+ lluvia1T[i-1] +" pulgadas");
            System.out.println("Trimestre 2: "+ lluvia2T[i-1] +" pulgadas");
            System.out.println("Trimestre 3: "+ lluvia3T[i-1] +" pulgadas");
            System.out.println("Trimestre 4: "+ lluvia4T[i-1] +" pulgadas");
            System.out.println("-----");
        }
// Muestra el promedio de Lluvia por trimestre en todos Los años
        System.out.println("El promedio de lluvia para el trimestre 1 durante " + anios+ " años es de: "+ promedio1T +" pulgadas" );
        System.out.println("El promedio de lluvia para el trimestre 2 durante " + anios+ " años es de: "+ promedio2T +" pulgadas" );
        System.out.println("El promedio de lluvia para el trimestre 3 durante " + anios+ " años es de: "+ promedio3T +" pulgadas" );
        System.out.println("El promedio de lluvia para el trimestre 4 durante " + anios+ " años es de: "+ promedio4T +" pulgadas" );
        System.out.println("-----");
// Muestra el promedio anual de Lluvia
        System.out.println("El promedio de lluvia anual durante " + anios+" años es de: "+ promedio +" pulgadas" );

// Pregunta al usuario si desea ingresar más información
        System.out.println("¿Desea ingresar más información? (s/n)");
        String respuesta = input.next();
        if (respuesta.equals("s")) {
            bandera = true; // Repite el ciclo si la respuesta es 's'

```

```
        }
    }  
    else{  
        System.out.println("Gracias por usar nuestro sistema, ¡Hasta  
pronto colega!");  
        bandera = false; // Termina el ciclo si la respuesta es  
diferente  
    }  
} while (bandera == true); // Ciclo principal del programa  
}  
  
// Función para calcular el promedio de lluvia de un trimestre  
public static double calcularPromedio(double[] lluviaTrimestre) {  
    double suma = 0;  
    for (double lluvia : lluviaTrimestre) {  
        suma += lluvia;  
    }  
    return suma / lluviaTrimestre.length;  
}  
  
// Función para calcular la suma total de lluvia de un trimestre  
public static double suamTotal(double[] lluviaTrimestre) {  
    double suma = 0;  
    for (double lluvia : lluviaTrimestre) {  
        suma += lluvia;  
    }  
    return suma;  
}  
}
```